Exam 2: Mystery Multivariate Data

36-402, Advanced Data Analysis

Due at 5 pm on Tuesday, 12 April 2011

Please read the background section, and all of the questions, carefully before beginning to work.

You will be sent a data set (CSV format) by e-mail to your Andrew account. Each data set is slightly different. Work only with your own. The origin f your data has been suppressed, so all columns are merely named "X.1" through "X.10", and the rows are just named from 1 to 1000. If you have not received a data set, or cannot open it, or it has the wrong format, contact Prof. Shalizi by 9 am on Wednesday, 6 April. If you do not do so, the presumption will be that you have received and can read your data.

You must turn in both a written response to the questions, and all of your supporting R code.

Turn in a hard-copy of the write-up to Prof. Shalizi, either in his office (Baker Hall 229C) or in his mailbox in the statistics department (Baker Hall 232). Include a signed copy of the last page of this exam as a cover sheet.

Turn in your code by uploading a plain text file to Blackboard. Name the file `andrewID-2.R`, where of course `andrewID` is your actual Andrew username. Make sure the file is in plain text format, so that it can be loaded into R and run; files in other formats will not be graded. Please submit your code only once; if you submit multiple versions, there is no guarantee that the version we grade will be the latest one. (Appeals to change grades on that basis will be denied.) Please do *not* submit your write-up electronically.

All work must be submitted by 5 pm on Tuesday. If you have not been able to finish the exam by that point, please turn in whatever you have done, for partial credit. Late exams will get no credit.

Many questions ask you to explain, describe or comment. Since communication is an essential part of data analysis, you *will* be graded on your writing. Be clear, be concise, and use your own words.

Note that the answers to some questions are very different for different data sets.

1. *Initial exploration* (15 points)

   (a) (4 points) Check whether the marginal distribution for each variable is Gaussian; both graphical and quantitative tests are acceptable.

   (b) (5 points) Explain what the test you used in problem 1a does, and why it works.

   (c) (3 points) Explain a *different* procedure that you could have used.

   (d) (3 points) Explain why making a scatter-plot of variable values against row numbers would *not* let you check whether a distribution is Gaussian.

2. *A joint distribution* (15 points)

   (a) (4 points) Using `npudens` from the `np` package, or any similar function, make a kernel density estimate of the joint distribution of `X.1` and `X.10`. Plot it; contour, color or perspective plots are all acceptable.

   (b) (5 points) Fit a two-dimensional Gaussian to the same data, and plot it in the same way.

   (c) (1 point) Plot the difference between the non-parametric and parametric density estimates. Comment.

   (d) (5 points) Could you use the same procedure as in Problem 1 to check that the joint distribution is Gaussian? If so, explain how to modify it and why it still works. If not, explain why it cannot be adjusted, and describe a different procedure which you could use. EXTRA CREDIT (10 points): implement your test and report your results.

3. *Principal components* (15 points)

   (a) (10 points) Find the first five principal components. Make a plot of these components. The horizontal axis should run over the integers from 1 to 10, the vertical axis should run from -1 to $+1$, and the points should indicate the projection of the components on to the corresponding observable variables. Put all five components in the same plot, using color or line style to distinguish them. (Alternately, use a three-dimensional plot.) Make sure the results come through clearly when printed. Comment on any patterns you see in the components. Grading *will* reflect how much visual clarity you can give this plot.

   (b) (5 points) Plot the amount of variance retained by the first q components vs. q, for up to 10 components.

4. *Factor analysis* (15 points)

(a) (5 points) Fit a factor model with one factor. Plot the factor loadings of the ten observable variables, as in the previous problem. Does this match the first principal component? Should it?

(b) (5 points) Fit a two-factor model, and plot the loadings of both factors. Has the first factor changed? If so, what does this mean? If not, is this a coincidence, or should the first factor never change when other factors are added?

(c) (5 points) Plot the amount of variance retained by the first q factors vs. q, for up to 5 factors. Does this match the variance plot for PCA (up to five components)? Should it?

5. *Factor model selection* (20 points)

(a) (7 points) Describe how to use the log likelihood ratio to select the number of factors.

(b) (3 points) Of the models fit in the previous question, which one is favored by the log likelihood ratio test?

(c) (7 points) Describe a way to test whether the discrepancy between your selected factor model and the data is significant.

(d) (3 points) Is the discrepancy significant?

6. *Mystery R* (20 points)

```
myfunction <- function(x, t=100, eps=1e-2/sqrt(nrow(x))) {
  stopifnot(require(mixtools),is.array(x))
  n <- nrow(x)
  w <- rep(c(0,1),length.out=n)
  w <- sample(w)
  del <- Inf
  i <- 0
  while ((del > eps) && (i < t)) {
    i <- i+1
    l1 <- mean(abs(w))
    l2 <- 1 - l1
    m1 <- apply(x,2,weighted.mean,w=w)
    m2 <- apply(x,2,weighted.mean,w=(1-w))
    s1 <- cov.wt(x,wt=w)$cov
    s2 <- cov.wt(x,wt=(1-w))$cov
    p1 <- dmvnorm(x,m1,s1)
    p2 <- dmvnorm(x,m2,s2)
    wn <- l1*p1/(l1*p1+l2*p2)
    del <- max(abs(wn-w))
    w <- wn
  }
  return(list(m1=m1,m2=m2,s1=s1,s2=s2,l1=l1,l2=l2,w=w,t=i,del=del))
}
```

(a) (6 points) Explain what this function does. What are the inputs? What are the outputs?

(b) (6 points) Explain what each line does.

(c) (4 points) Is the `abs` necessary in the second line of the `while` loop? Is it necessary in the next-to-last line of the `while` loop?

(d) (4 points) Run this function on your data. Describe the results.

7. *Mixtures* (Extra Credit, 50 points) Install the `mixtools` package from CRAN, and read sections 1, 2, and 6.1 of the paper describing it by Benaglia *et al.* (`http://www.jstatsoft.org/v32/i06`).

   - (10 points) Using `mvnormalmixEM()`, fit Gaussian mixture models to your data, varying the number of clusters or mixture components from 2 to 6. Plot the likelihood as a function of the number of clusters.

     Hint: The default settings for `maxit` and `epsilon` will take forever; more reasonable ones here would be `maxit=100` and `epsilon=1e-1`. Explain, in your write-up, what these settings mean. Fitting the model with seven clusters might then still take up to an hour on a slow machine.

   - (10 points) Describe how the `boot.comp` function works.

   - (10 points) Use `boot.comp` to select the number of components to use. *Hint*: You will want to use fewer than the default number of bootstrap replicates, and also pass along the fitting arguments. Even so, this may take a very long time.

   - (10 points) Describe a way to decide whether to use this selected mixture model, or the factor model you selected earlier, and explain why this comparison should be reliable.

   - (10 points) Implement your comparison. Which model is favored?

36-402, Advanced Data Analysis, Spring 2011

Examination 2

I have read the university policy on cheating and plagiarism (`http://www.cmu.edu/policies/documents/Cheating.html`). I have completed this take-home examination honestly, without giving or receiving prohibited assistance to anyone.

SIGNED:

NAME: